Asian Journal of Information Technology 17 (4): 261-270, 2018
ISSN: 1682-3915

Improvising Classification Performance for High Dimensional and Small Sample Data Sets

[1]L. Kamatchi Priya, [1]M.K. Kavitha Devi and [2]S. Nagarajan
[1]Vickram College of Engineering, Sreenivasa Gardens, Madurai Sivagangai Road, Enathi, 630561 Tamil Nadu, India
[2]Thiagarajar College of Engineering, GST Road, Thiruparankundram, Madurai, 625005 Tamil Nadu, India

Abstract: Classification is an important problem where the performance of a classifier depreciates as the sample size decrease and dimensionality increase. This study describes feature subset selection framework for supervised classification problem which works efficiently with very few training samples. In the proposed algorithm, the most relevant feature has been selected by using filter method and the redundancy among the features is eliminated by using correlation-based spanning tree. The proposed framework is designed to perform data analytics to extract the most influencing predictors. The complexity of the algorithm is reduced drastically by performing parallel processing of feature subsets. The performance of the algorithm is tested against various predominant feature subset selection algorithms in 4 different datasets from UCI repository and 2 real world microarray data where the classification accuracy of the proposed framework is better than the others feature selection algorithms.

Key words: Feature subset selection, filter method, correlation-based spanning tree, supervised classification, datasets, proposed framework

INTRODUCTION

Dimensionality reduction problem has to be resolved for improving the performance of the machine learning techniques which are used for classification or regression. Dimensionality reduction can be performed by feature extraction or feature subset selection. Feature extraction techniques transform the data into low dimensional data, so that, useful information is extracted from all the features. Depending upon the availability of class or target information, they are classified as supervised and unsupervised.

Supervised feature extraction method like Fisher Linear Discriminant (FLD) extracts the most relevant discriminant vectors (Fisher, 1936). Un-supervised feature extraction methods like Principal Component Analysis (PCA) (Jolliffe, 2002), Locally Linear Embedding (LLE), (Roweis and Saul, 2000), kernel PCA (k-PCA) (Scholkopf *et al*., 1998) and Laplacian Eigen map (LE), (Niyogi, 2004; Belkin and Niyogi, 2002) protect the universal covariance structure of data when the targets are not known.

Most of the decision making managerial problems do not stop with predicting but are also, intended to know the variables or features which influence the output. In such cases, feature selection is preferred over feature extraction. Feature selection can also be categorized into supervised and unsupervised. Some of the supervised feature selection methods like Fisher score (Duda *et al*., 2012), Relief and Relief (Robnik and Kononenko, 2003), Fast Correlation-Based Filter (FCBF) (Lei and Liu, 2004) and Spectrum decomposition (SPEC) (Zhao and Liu, 2007), evaluate the relevance of feature with the target class labels. Unsupervised feature selection is performed using variance score (Bishop, 1995), Laplacian score and Hilbert-Schmidt Independence Criterion (HSIC) (Song *et al*., 2012; Yamada *et al*., 2014).

Feature subset selection can also be classified as Filter method and Wrapper method. Filter methods are independent of machine learning techniques, used for classification or prediction (Sanchez *et al*., 2007). In wrapper methods, feature ranking technique is wrapped around machine learning techniques. Most of the feature subset selection problems are addressed by wrapper method whose performance is better than filter method (Talavera, 2005; Kohavi and John, 1997). But wrapper method requires large training set. For the problems with very few instances wrapper method is infeasible and the performance of the feature selection deteriorates as the ratio between the number of attributes to the number of

Corresponding Author: L. Kamatchi Priya, Vickram College of Engineering, Sreenivasa Gardens, Madurai Sivagangai Road, Enathi, 630561 Tamil Nadu, India

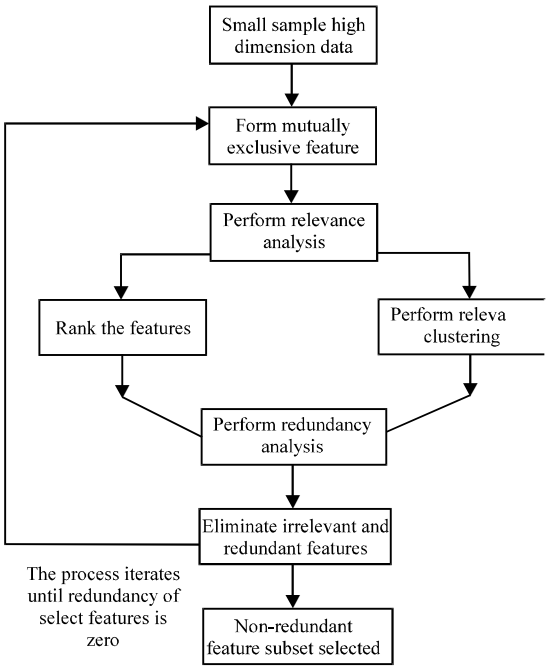

Fig. 1: A feature sub-set selection framework for diminutive dataset

instances increases (Kohavi and John, 1997). The chore of feature subset selection becomes tougher when there are very few instances to train for a high dimensional data. On the other hand, most of the supervised filter method finds the relevance between the features and the class label but fails to remove the redundant features. Redundant feature augments the complexity of learning and decrease the performance (Peng *et al*., 2005). Thus, feature relevance with the target and redundancy among the features are the 2 major concerns in dimensionality reduction problems.

In this proposed research, the designed framework is for small sample size referred as diminutive training set with large number of features to perform feature subset selection in Fig. 1. When the number of training instances is less than or equal to one tenth of its dimension, then it is called as diminutive training set. This framework consists of three functions for selecting best subset of features. The first one ranks the features based on the restraint fisher score with the target labels. The second one cluster the redundant features based on the relevance between the features using correlation based on maximum spanning tree. Finally, the last function chooses the most relevant features which are totally independent of each other thus removing redundancy by using the relevance clusters. The feature subset selection algorithm is capable of reducing the features without the assistance of an optimization algorithm to choose the number of features by utilizing Recursive Redundancy Elimination (RRE) for diminutive training set, thus, reducing the complexity of the algorithm drastically. Small sample data with high dimension requires feature subset selection. In addition, even if the features selected are relevant to the target vector, redundant features can significantly increase the complexity of learning and generally depreciate classification accuracy.

MATERIALS AND METHODS

Feature subset selection for diminutive training data: This study describes some definitions and notations used for feature subset selection in a small dataset context.

Definitions and notations: For small sample learning, a data set of N instances, $X = \{x_1, \ldots, x_N\}$ and M features of X are $F = \{f_1, f_2, \ldots, f_M\}$ be the corresponding feature vectors that record the feature value in each instance. The corresponding target vector or labels are given by $Y_N = \{y_1, \ldots, y_N\}$ where $N<<M$. The feature vector F is divided into U subsets of size h where h is not greater than N ($h<N$) such that $F = \{\{f(1, 1), \ldots, f(h, h)\}, \{f(h+1, 1), \ldots, f(2h, h)\}, \ldots, \{f(Uh+1, 1), \ldots, f(N, h)\}\}$ or $F = \{F_1, \ldots, F_U\}$ where $F_i = \{f_{i+1}, \ldots, f_{i+h}\}$ and I $\{0, 1, \ldots, U-1\}$.

Restraint relevance clustering: For a given set of features, relevance clustering is performed to form m clusters. In each cluster Ω_{xi}, any pair of features f_r and f_j, the different type of restriction are:

- Must-Associate constriction (MA): relating f_r and f_j, specifies that they are highly correlated
- Cannot-Associate constriction (CA): relating f_r and f_j, specifies that they are independent of each other

MA and CA constraints subdivide the subset Ω_x into $\Omega_{x1}, \ldots, \Omega_{xm}$ clusters. The features are selected such that no two features in Ω_{xi} are elected to resultant feature subset.

Related work: In this study, we discuss the feature scoring and redundancy elimination through graph theory with their limitations.

Fisher score: This score is used for supervised feature selection. The Fisher score inspects the distance between the data points in the same class. These data points are as close as possible and distance between the data points in the different class are as far away as possible and ranks the feature accordingly. The fisher score F (Z) is computed for each feature based on the relevance with the target vector. The Fisher score is computed as follows:

$$F(Z) = S_b (S_t + \gamma I)^{-1} \quad (1)$$

Where:

$\tilde{S}_b$ = Between-class scatter matrix
$\tilde{S}_t$ = Total scatter matrix
γ = A positive regularization constraint
I = Identity matrix S_b and S_t

Are defined as:

$$\tilde{S}_b = \sum_{K=1}^{c} n_k (\tilde{\delta}_k - \tilde{\delta})(\tilde{\delta}_k - \tilde{\delta})^T \quad (2)$$

$$\tilde{S}_t = \sum_{i=1}^{c} (\tilde{\varepsilon}_i - \tilde{\delta})(\tilde{\varepsilon}_i - \tilde{\delta})^T \quad (3)$$

Where:

$\tilde{\delta}_k$ and n_k = The mean vector and size of the kth class, respectively in the reduced data space Z
$\tilde{\delta} = \sum_{k=1}^{c} n_k \tilde{\delta}_k$ = The overall mean vector of the reduced data
$\tilde{\varepsilon}_i$ = The mean of individual feature vector and c represents the dimension of the data

Maximum spanning tree based redundancy elimination: Maximum spanning tree is constructed to eliminate the maximum number of redundant features and keep the strong relevant ones. This method decides on h superlative features and build the graph $G_h (V_h, E_h)$ where $G_h(V_h, E_h)$ is a weighted graph where V is the set of vertices representing h relevant features and E_h is the set of edges connecting h vertices. An edge weight represents the correlation between vertices (features) which are connected by the edge. Maximum spanning tree $G'_h(V'_h, E'_h)$ is built from G_h using Prim's or kruskal's algorithm. Relevant feature Fr is chosen from V'_h such that $(F_r, F_j)E'_h$. This procedure is repeated for all relevant features. G is the adjacency matrix of size h×h representing the graph $G_h(V_h, E_h)$ (Benabdeslem and Hindawi, 2014) has used minimum spanning tree to form clusters but in different contexts.

Discussion and motivation: Fisher score computes the score for each feature which is highly relevant to the target variable but fails to consider the combination of the features. That is to say, the relevance of individual features is low but the combination of various features is high. Secondly, they cannot eliminate redundant features. Thus, handling redundant features is mandatory while using fisher score.

Maximum spanning tree based redundancy elimination technique proved to be the best for feature selection for classification and clustering (Benabdeslem and Hindawi, 2014) but the complexity of algorithm increases as the h increases. Moreover, determining h is done either by an optimization algorithm which adds up the complexity or choosing a random number which may pay no heed to a relevant feature thus tumbling the performance of the system.

Proposed approach: In this study, we present our feature subset selection framework for diminutive feature selection. It incorporates a restraint Fisher score and relevance clustering by using correlation based maximum spanning tree. In addition it exploits a recursive redundancy elimination algorithm for more proficient feature selection in the diminutive training context.

Restraint feature ranking using Fisher score: Filter-based feature selection method like Fisher score is usually cast into a binary selection of features which maximizes the performance criterion (Benabdeslem *et al*., 2014). Fisher score determines the relevance between a feature fr and the class labels Y. It picks each feature autonomously according to, their scores determined using Fisher criterion. Fisher weights of the rth feature are given as follows (Duda *et al*., 2012):

$$W(X^r) = \frac{\sum_{k=1}^{c} n_k (\delta_k^r - \delta^r)^2}{(\sigma^r)^2} \quad (4)$$

Where:

δ^r and σ^r = The mean and standard deviation of the whole data set X, corresponding to rth feature
δ_k^r, σ_k^r and n_k = The mean, standard deviation and size of the kth class, corresponding to the rth features

The Fisher score for the rth feature is given as Fr.

$$F_r = \begin{cases} 0, \text{if} \sum_{k=1}^{c} n_k (\delta_k^r - \delta^r)^2 \text{ is zero} \\ \infty, \text{if} (\sigma^r)^2 \text{ is zero} \\ W(X^r), \text{Otherwise} \end{cases} \quad (5)$$

Sort the features in non decreasing order of their Fisher score. Then rank the features such that the feature with higher score has the least rank. Φ has the features and their rank:

$$\Phi_i = \{i^{th} \text{ minimum of } F_r \quad (6)$$

Feature relevance clustering: In this study, we propose a new clustering technique which assists in removing the redundant features as a part of feature subset selection.

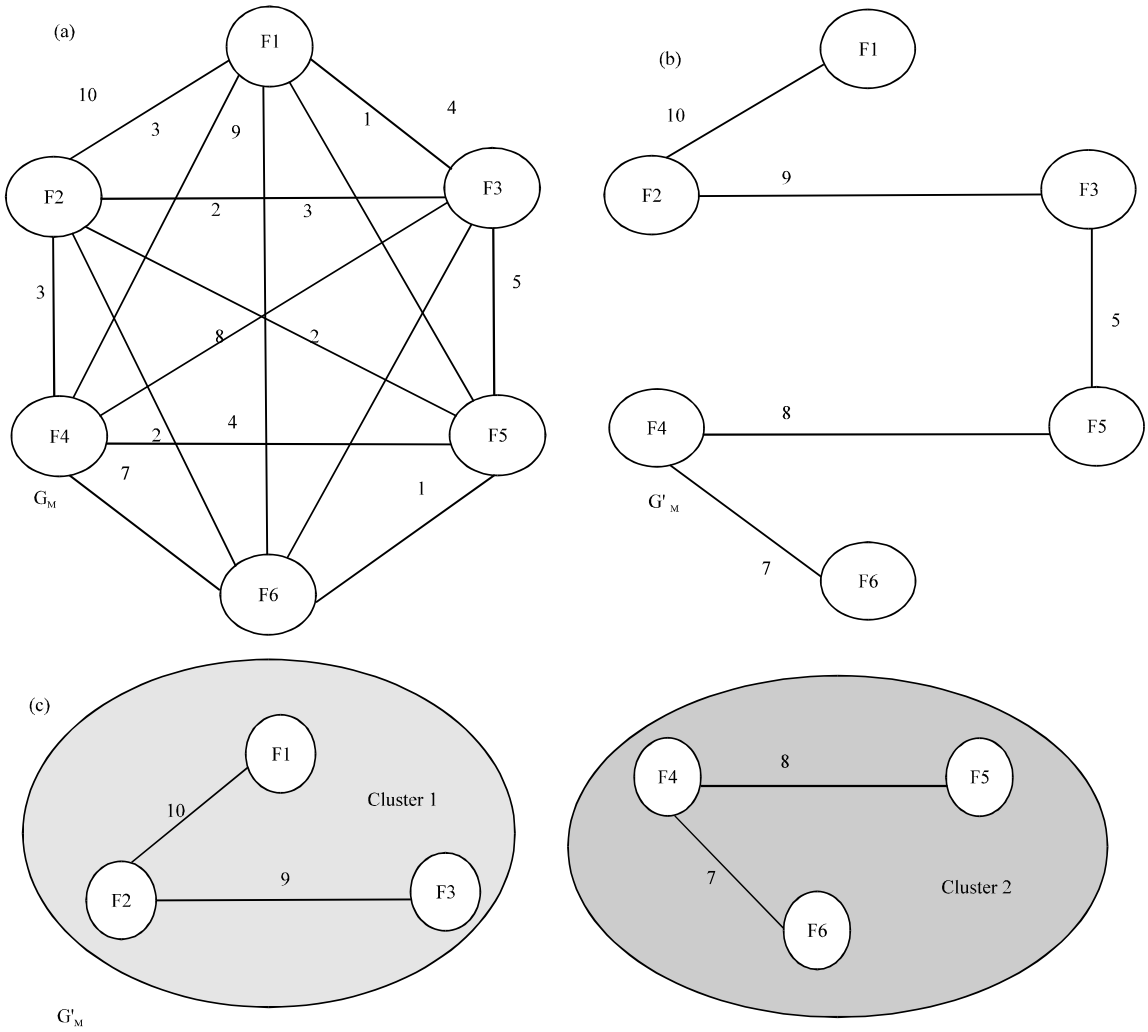

Fig. 2: a) GM original graph where each node represents a feature. And the edge connecting the nodes represent correlation between the features; b) G'$_M$ maximum spanning tree for the graph G_M and c) G"M relevance clusters formed by removing minimum edge

Correlation measure: The predominant measure which is used to find the relationship between two features f_r and f_c is correlation coefficient. It is defined as follows (Schreiber *et al*., 2003):

$$(f_r, f_c) = \frac{\sum_i (f_{ri} - \bar{f}_r)(f_{ci} - \bar{f}_c)^2}{\sqrt{\sum_i (f_{ri} - \bar{f}_r)^2} \sqrt{\sum_i (f_{ci} - \bar{f}_c)^2}} \quad (7)$$

where, $\bar{f}_r$ and $\bar{f}_c$ are the means of the feature vector f_r and f_c, respectively. Correlation matrix R contains pair wise correlation coefficient between each pair of features in the input dataset. R represents the feature-feature relevance and the strength of their association. The features which are highly correlated are extremely dependent or extremely relevant to each another. The matrix R is given as follows:

$$R_{\downarrow(r,c)} = \begin{cases} 0, \text{if} \sum_{\downarrow i} \cong \left(f_{\downarrow ri} - \left(f_{\downarrow r}\right)^-\right)\left(f_{\downarrow ci} - \left(f_{\downarrow c}\right)^-\right) \text{is zero} \\ @1, \text{if} \left(\sum_{\downarrow i} \cong \left(f_{\downarrow ci} - \left(f_{\downarrow c}\right)^-\right)^{\uparrow 2}\right) \text{is zero} \end{cases} \quad (8)$$

Relevance clustering using maximum spanning tree: In this study, we intend to cluster the features which are highly correlated using graph based method. We propose a strategy to cluster relevant features by using correlation based maximum spanning tree. This technique requires an adjacency matrix representing a graph $G_M(V_G, E_G)$ whose vertices V_G are the features, edge cost E_G depicting the absolute value of correlation between the features (i.e., E_G = abs ($R_{r,c}$)) and G_M is manipulated as follows (Fig. 2):

$$G_M(r,c)=\begin{cases}E_G, \text{if abs}(R_{r,c})>0\\ 0, \text{Otherwise}\end{cases} \quad (9)$$

The maximum spanning tree G'_M is a connected acyclic sub-graph of G_M such that the sum of edge cost is maximum. This is constructed using an optimization prim's algorithm (Stoer and Wagner, 1994). Figure 2 a represents a complete graph where all the edges are weighted using the correlation values (number of features, M = 6). Where Fig. 2b depicts maximum spanning tree G'_M obtained from G_M where the edge connecting the vertices represent highest correlation among the features considered:

$$G'_M(r,c)=\begin{cases}E_G \text{ where } E'_G = \max_M(E_G)\\ 0, \text{Otherwise}\end{cases} \quad (10)$$

Threshold value is determined by calculating the mean of R. and is defined as follows:

$$\overline{R}_{r,c}=\frac{\sum_i^r\sum_j^c R(i,j)}{n^2} \quad (11)$$

i and j are iterative variables which varies from 1-r and c respectively. Where r and c represents number of rows and columns of R. Remove all the edges whose cost lesser than the threshold value $\overline{R}_{r,c}$. The nodes or vertices or features which are connected form a cluster. Figure 2c depicts the cluster of features.

$$E_{\downarrow G'}(r,c)=\begin{cases}E_{\downarrow G'}(r,c), \text{where } E_{\downarrow G'(r,c)}>\\ R^{-}_{\downarrow}(r,c)@\ 0, \text{Otherwise}\end{cases} \quad (12)$$

$$G'_M(r,c)=\{(1, E_{\downarrow G''(r,c)}>0\ @\ 0, \text{Otherwise}) \quad (13)$$

Algorithm 1; Restraint Relevance Clustering (RRC):

Input : Feature Subset Ω_x, for Input X_N
Output : Forest G''= {$\Omega_{x1},\ldots,\Omega_{xm}$}, 'm' feature clusters
1: Construct the Correlation matrix R from Ω_x using Eq. 8
2: Construct the graph $G_M(V_M, E_M)$ where each V_M represent the features in $\Omega_x$and E_M represents the correlation between the features in Ω_x using Eq. 9
3: Find the maximum spanning tree $G'_M(V_M, E_M)$ from G_M using Prims's
4: Calculate threshold value $\overline{R}_{r,c}$ using Eq. 11
5: Remove the edges whose cost lesser than the threshold value
6: Identify the features which are connected as a cluster
7: Return the 'm' relevance clusters G''= {$\Omega_{x1},\ldots,\Omega_{xm}$}

Feature elimination: Feature elimination is done such that features are non redundant and highly relevant to the target vector. In this study we present paramount restraint feature elimination and recursive redundancy elimination for diminutive dataset with massive features.

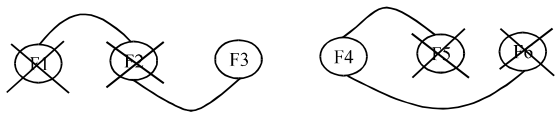

Fig. 3: Elimination of redundant features (from G''_M in Fig. 2c)

Table 1: Ranking of features shown in Fig. 3

Feature rank	Feature number
1	F3
2	F2
3	F4
4	F1
5	F5
6	F6

Paramount restraint feature elimination: In this study, we will choose features to eliminate redundancy such that the element is highly relevant to the target vector. Feature redundancy is directly allied to feature correlation. It is broadly acceptable that 2 features are redundant to one other if their values are entirely correlated (Lei and Liu, 2004; Benabdeslem and Hidawi, 2014). The relevant clusters formed by relevance clustering using algorithm-1 imply all the features connected are highly redundant. Feature subset selection is implemented such that a feature is selected from a cluster which is highly relevant with the target vector given by the feature ranking algorithm which uses Fisher score. Sort the features based on the rank in non decreasing order such that the least rank feature is highly relevant with the target vector. One feature with least rank is selected from each cluster and the rest is eliminated and is shown in Fig. 2c and feature ranking for the features in Fig. 2 is shown in the Table 1 where M = 6.

Algorithm 2; Paramount Restraint Feature Elimination-PRFE:

Input: Feature Subset G_2= {$\Omega_{x1},\ldots,\Omega_{xm}$}Feature Ranking Φ_x
Output : F_{xi}={$f_1,\ldots, f_m$}, m relevant non redundant features
1: Rank the features using Eq. 5
2: Sort the features based on their rank in Φ_x where Φ_x={$f_1, \ldots, f_n$} such that rank(f_1) < rank (f_2) <, … , < rank(f_n)
3: repeat
4: Select the relevant feature (f_i,f_j) from Φ_x, such that for all (f_i,f_j) in F_x (f_i,f_j) does not belong to the same cluster {$\Omega_{x1}, \ldots, \Omega_{xm}$}
5: And rank (f_{xi})<min(rank(f_r)) where $f_r \Omega_{xi}$
6: until no more features can be selected from Φ_x
7: return F_x

Recursive redundancy elimination for diminutive training set: For dataset with a very small number of instances and a large feature set cannot employ the wrapper method, since, the learning performance depreciates as the instance to feature ratio decreases. The major limitation of feature subset complexity of algorithm

is propositional to the number of features and if there are millions of features it would be a time consuming to compute all the pairwise correlations. To address this issue the features can be split into subsets and moreover this method will not affect the performance, since, Fisher score determines the relevance between an individual feature and the target and not the combination. This technique is beneficial in improving the complexity of the algorithm. While finding the correlation matrix, the dimension of the matrix is reduced drastically and thus the complexity of the algorithm is reduced.

Algorithm 3; Recursive Redundancy Elimination (RRE) for diminutive training set:

Input : Input Dataset X_N and Target Class Labels Y_N
Output : $F_x=\{f_1,\ldots,f_d\}$, 'd' relevant non redundant features
1: repeat
2: Break up the input data into 'n' mutually exclusive feature subsets each with 'h' features, such that h<N where N is the number of instances
3: For each subset Ω_{xi}, perform algorithm 1 to get 'm' relevant feature clusters
4: From each relevant cluster pick one feature, thus having 'm' features using algorithm 2
5: merge 'h' features from 'n' cluster to form a feature subset or input data for next iteration
6: until 'n' becomes one to completely remove redundancy
7: return F_X with 'd' features which are non redundant to each other and relevant to target class labels

Next important question is how to determine the number of features in a subset. Many studies have agreed that when the number of instance is greater than the features, the classification performance improves (Joachims, 1998). Hence, the number of feature in a subset h is chosen such that h<N; Usually N/2<h<N (i.e., lies between N/2 and N). The algorithm 1 and 2 are executed for every subset of features. And the feature subsets are mutually exclusive to each other. This is an iterative process, since, the features selected from each subset serves as the input to the next iteration and the process repeats until redundancy is removed completely (i.e., n = 1). Finally a proper feature subset selection is performed where the selected features are non redundant to each other and relevant to the class labels. This algorithm is self-governing and does not wait for input from the user like number of features to be selected.

RESULTS AND DISCUSSION

Experimental study: In this study, we pragmatically assess the performance of the algorithm derived from the RRE framework (algorithm 3). The study is made in the Diminutive training context with redundancy analysis.

Data sets and methods: In the research, 4 high dimensional data sets (M>300) with very small training set (N = 10, N<M/10) is selected. The datasets are available in UCI machine learning repository. The whole data set information is detailed in Table 2 in which the last column represents the ratio of instances to features. This is shown to evident that feature selection algorithm performs better (higher prediction accuracy) for very small dataset S<0.1). The data sets are high-dimensional with very small training set.

Table 2: Training instance to feature ratio

Data sets	N	M	S = N/M
LSVT	10	309	0.0324
Arcene	10	10000	0.0010
Madelon	10	500	0.0200
Dorothea	10	5576	0.0018

Table 3: Data set description

			Training set		Test set	
Data sets	Actual features	Class values	N_{tr}	Percentage	N_{ts}	Percentage
LSVT	309	0	6	60	24	66.67
		1	4	40	42	33.33
Arcene	10000	0	6	60	56	56
		1	4	40	44	40
Dorothea	5576	0	7	70	722	90.25
		1	3	30	72	9.75
Madelon	500	0	5	50	1000	50
		1	5	50	1000	50

The problem is to train with very small data (say 10 instances) Table 3 presents the number of instances used in the training data (N_{tr}) and test data (N_{ts}). The distribution of examples belonging to class 0 and class 1 are shown (both count and percentage are logged). This clearly shows that test data has varying distribution. It is evident that if an algorithm can perform better for all these data, then it is feasible for any given data. To evaluate the performance of RRE algorithm and to compare it with other methods, we choose 4 representative methods.

Fisher, commonly referred as Fisher score is basically a supervised feature selection method that can be used to find the relevance between two vectors (Peng *et al*., 2005).

mRMR, minimum Redundancy-Maximum Relevance is a Filter method which removes the redundant features using mutual information between the features (Maji and Paul, 2011; Zeng *et al*., 2014). SBMLR, Sparse Bayesian Multinomial Logistic Regression calculates the weights for a sparse multinomial regression model where the sparsely populated data is obtained using Bayesian regularization with a Laplace prior (Cawley *et al*., 2007).

RREDT, Recursive Redundancy Elimination for Diminutive Training set is the proposed framework which removes the redundant features using pair-wise feature correlation.

Table 4: Classification accuracy (%: higher the better)

Data sets	Actual features	Features selected	Prediction accuracy Fisher	mRMR	SBMLR	RREDT
LSVT	309	29	63.49% (3)	61.11% (4)	69.05% (2)	73.81% (1)
Arcene	1000	23	61% (2)	54% (4)	60% (3)	64% (1)
Madelon	500	43	50.55% (4)	51% (2)	51% (2)	51.1% (1)
Dorothea	5576	17	80.13% (2)	61.5% (4)	73.63% (3)	82% (1)
Average Rank			2.75	3.5	2.5	1

Classification is done after feature selection. Since, very few instances are available for training, Bootstrap aggregation ensemble decision trees are used commonly known as TreeBagger. Every tree in the ensemble is grown on an independently drawn bootstrap replica of input data to improve classification accuracy. Bootstrap aggregating, also, called bagging is a machine learning ensemble algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification. It also, reduces variance and helps to avoid over fitting. Moreover, it performs well for small dataset.

Experimental setting: Each data set is split into training partition with 10 instances randomly chosen and a test partition with the remaining instances. After feature selection, TreeBagger classifier is employed for classification. Accuracy is determined using the following:

$$acc=\frac{\alpha}{\alpha+\beta}\times 100 \quad (14)$$

where, α denotes number of classes identified correctly, β denotes number of classes identified wrongly and $\alpha+\beta$ is the total number of instances. The classification is performed several times and the average accuracy is tabulated in Table 4. Finally, for redundancy analysis, we used the same measure used by Zhao *et al.* (2011):

$$redu(F)=\frac{1}{M(M-1)}\sum_{F_i,F_j\in F,\ i>j}\rho(F_i,F_j) \quad (15)$$

where, F is the final set of selected features, $\rho_{i,j}$ returns the Pearson correlation between the two features F_i and F_j. The redundancy measurement evaluates the average correlation among all feature pairs. Large value indicates the features are strongly correlated and thus redundancy is expected to exist in F.

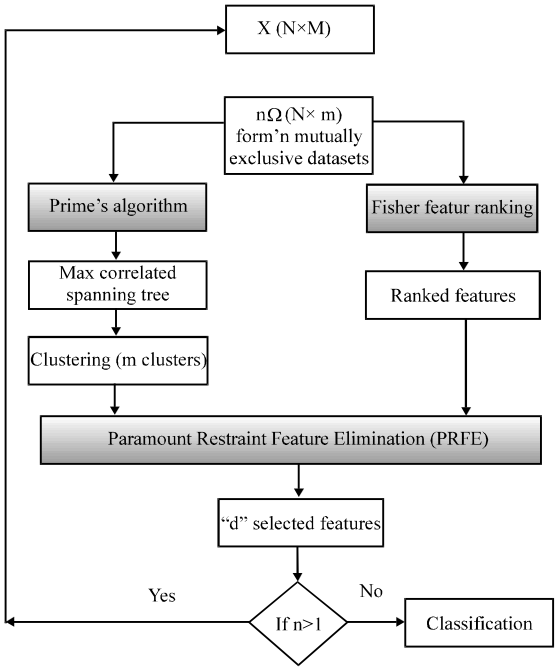

Fig. 4: Feature subset selection framework RRE

Feature quality on classification performance: In this study, we evaluate the performance of our framework and compare it with other feature selection methods. This comparison concerns the classification accuracy results that we present in Table 4. Which presents the prediction accuracy evaluated using Eq. 14 and the ranks for the feature selection algorithms pertaining to a particular dataset are given within the brackets. These ranks are used to perform Friedman test. Where the null hypothesis (H_0) is defined as "There is no proposition between the methods" and the alternate hypothesis (H_1) as "There is proposition between the methods and the true value indicates the proposition". The number of feature selected is also, tabulated. RREDT algorithm automatically converges to determine the number of features and the same is imputed to other algorithms to compare their performance. The true value (prediction accuracy in %) of RREDT algorithm is greater than other techniques compared. Table 4 clearly depicts the RREDT algorithm outperforms the other predominant feature selection algorithms (Fig. 4).

The speciality of the proposed algorithm is that it automatically converges to select the number of features required for classification. Whereas for other feature subset selection techniques the number of features is decided by the user or by the optimization algorithm. The decision of optimization algorithm increases the time complexity, since, it involves more iteration in selecting the optimal number of features. Table 4 shows the

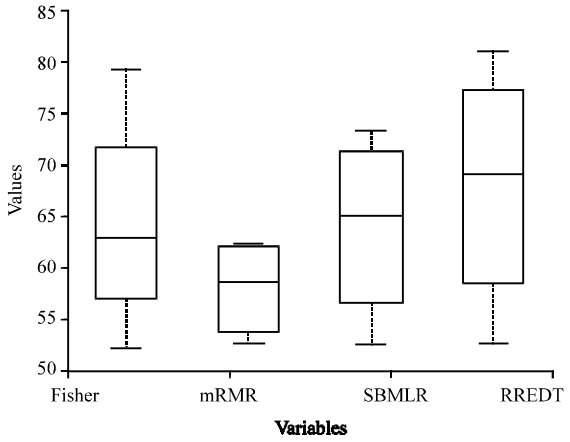

Fig. 5: Boxplot showing the statistical differences between the prediction accuracy of the combination of Fisher score (Fisher) minimum Redundancy-Maximum Relevance (mRMR) Sparse Bayesian Multinomial Logistic Regression (SBMLR) and Recursive Redundancy Elimination for Diminutive Training (RREDT) feature selection methods with treeBagger classification

number of features selected by RRE algorithm and the same number of features is used to test the performance of other algorithms. And it is found that results (selection of number of features) are same as that of optimization algorithm.

Figure 5 shows the box plot where x-axis has various feature selection techniques compared and y-axis denotes the prediction accuracy. The box represents the range of accuracy produced by the techniques and the line in between the box represents the mean accuracy. It can be stated that proposed system is more accurate than other feature selection algorithms. These assert is supported by a statistical test where the hypothesis "There is no proposition between the methods" was rejected at the 95% confidence level (Friedman test, value = 0.031, alternative hypothesis: there is proposition between the methods and the true value indicates the proposition).

In real world applications, the microarray data has very small number of training instances compared to its dimension. Two microarray data GLI-85 and TOX-171 from http://featureselection.asu.edu/datasets.php are analyzed with different feature subset selection algorithms and Table 5 and Fig. 6, clearly depicts that the proposed algorithm performs better than other feature subset algorithms in terms of classification accuracy. The microarray dataset GLI-85 has 85 instances and TOX-171 has 171 instances where the instances were equally

Table 5: Redundancy measure (smaller the better)

Data sets	Fisher	mrMR	SBMLR	RREDT
LSVT	8.00	0.03	3.46	0
Arcene	10.60	0.09	5.75	0
Madelon	6.96	0.27	4.75	0
Dorothea	0.70	0.10	6.85	0

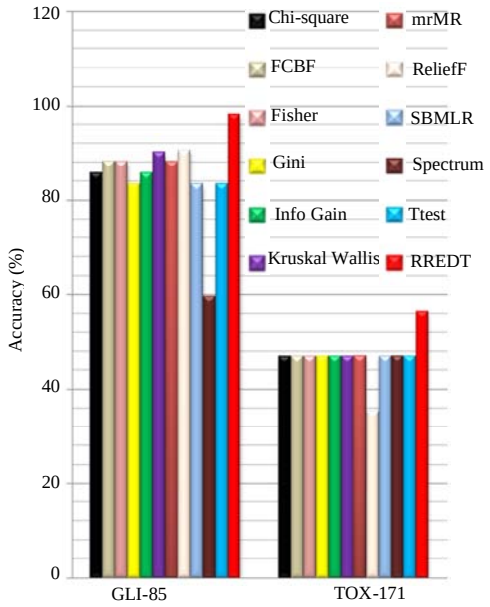

Fig. 6: SVM Classification accuracy of microarray dataset obtained from various feature subset algorithms

partitioned into two, one for training and other for testing. Similar analysis on small sample size problem were performed in Li *et al.* (2015) utilizes same dataset.

Feature quality on redundancy rate: In Table 6, the redundancy rates of feature subset selected for classification by different algorithms is shown for different data sets. Note that this number of features selected n is automatically determined by RREDT algorithm. The proposed framework ensures zero redundancy and is compared with mRMR which handles redundancy. We report in the table the best result among the three variants for each data set. We can see in Table 4 that RREDT efficiently removes redundancy. For this task it outperforms Fisher, SBMLR and mRMR. When feature selection used in decision making, redundant features will mislead decisions.

In this study, the proposed feature subset selection framework is designed for high dimensional small sample size classification problems. The proposed framework is highly generic and can be used in many real world applications addressed by the recent research (Junttila *et al.*, 2015; Shaghaghi and Vorobyov, 2015;

Table 6: Performance comparison of different feature subset selection algorithms on Micro-Array data

	GLI-85			TOX-171		
Feature selection Techniques	Total number of features	Optimal number of features	Accuracy (%)	Total number of features	Optimal number of features	Accuracy(%)
Chi-square	22283	17	85.72	5748	11	47.06
FCBF	22283	18	88.10	5748	11	47.06
Fisher	22283	27	88.10	5748	11	47.06
Gini	22283	13	83.33	5748	11	47.06
InfoGain	22283	14	85.71	5748	11	47.06
KruskalWallis	22283	12	90.00	5748	11	47.06
mr MR	22283	13	88.09	5748	11	47.06
ReliefF	22283	13	90.48	5748	11	35.29
SBMLR	22283	12	83.33	5748	12	47.06
Spectrum	22283	37	59.52	5748	18	47.06
t-test	22283	15	83.33	5748	16	47.06
RREDT	22283	9	98.10	5748	7	56.47

Lu and Zoubir, 2015; Binol *et al*., 2015). The proposed framework works by redundancy elimination and relevance analysis for diminutive (small sample size) dataset. A new function was developed to eliminate the redundancy among the feature based on geometrical structure of data and relevance of the data with the class labels, hence, the feature subset selected are the optimal predictors or decision variables.

CONCLUSION

The proposed framework has several advantages. It does not require an optimization algorithm to decide the number of feature used for classification, since, the proposed framework automatically selects the feature subset. It exploits the pairwise feature correlation to remove redundancy among the features completely. It elects to choose the most relevant feature among the redundant cluster using graph notional approach

RECOMMENDATIONS

Future research may include residual management. A new feature could be derived from the eliminated features. One or more component is extracted which represent non redundant relevant part of the eliminated feature, thus, improving the performance of the classification.

REFERENCES

Belkin, M. and P. Niyogi, 2002. Laplacian eigenmaps and spectral techniques for embedding and clustering. Adv. Neural Inform. Process. Syst., 14: 485-491.

Benabdeslem, K. and M. Hindawi, 2014. Efficient semi-supervised feature selection: Constraint, relevance and redundancy. IEEE. Trans. Knowl. Data Eng., 26: 1131-1143.

Binol, H., G. Bilgin, S. Dinc and A. Bal, 2015. Kernel fukunaga-koontz transform subspaces for classification of hyperspectral images with small sample sizes. IEEE. Geosci. Remote Sens. Lett., 12: 1287-1291.

Bishop, C.M., 1995. Neural Networks for Pattern Recognition. Oxford University Press, Oxford, UK., ISBN: 9780198538646.

Cawley, G.C., N.L. Talbot and M. Girolami, 2007. Sparse Multinomial Logistic Regression via Bayesian l1 Regularisation. In: Advances in Neural Information Processing Systems. Bernhard, S., J. Platt and T. Hofmann (Eds.). MIT Press, London, England, ISBN: 978-0-262-19568-3, pp: 33-40.

Duda, R.O., P.E. Hart and D.G. Stork, 2012. Pattern Classification. 2nd Edn., John Wiley and Sons, New York, USA., ISBN: 0-471-05669-3, Pages: 637.

Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugen., 7: 179-188.

Joachims, T., 1998. Text categorization with support vector machines: Learning with many relevant features. Proceedings of the 10th European Conference on Machine Learning, Chemnitz, Germany, April 21-23, 1998, Springer, Berlin, Heidelberg, pp: 137-142.

Jolliffe, I.T., 2002. Principal Component Analysis. 2nd Edn., Springer-Verlag, New York, USA.

Junttila, V., T. Kauranne, A.O. Finley and J.B. Bradford, 2015. Linear models for airborne-laser-scanning-based operational forest inventory with small field sample size and highly correlated LiDAR data. IEEE. Trans. Geosci. Remote Sens., 53: 5600-5612.

Kohavi, R. and G.H. John, 1997. Wrappers for feature subset selection. Artif. Intell., 97: 273-324.

Lei, Y. and H. Liu, 2004. Efficient feature selection via analysis of relevance and redundancy. J. Mach. Learn. Res., 5: 1205-1224.

Li, Y., J. Si, G. Zhou, S. Huang and S. Chen, 2015. FREL: A stable feature selection algorithm. IEEE. Trans. Neural Netw. Learn. Syst., 26: 1388-1402.

Lu, Z. and A.M. Zoubir, 2015. Source enumeration in array processing using a two-step test. IEEE. Trans. Signal Process., 63: 2718-2727.

Maji, P. and S. Paul, 2011. Rough set based maximum relevance-maximum significance criterion and gene selection from microarray data. Int. J. Approximate Reasoning, 52: 408-426.

Niyogi, X., 2004. Locality Preserving Projections. In: Neural Information Processing Systems. Sebastian, T. and K.S. Lawrence (Eds.). Massachusetts Institute of Technology, Cambridge, Massachusetts, pp: 153-160.

Peng, H., F. Long and C. Ding, 2005. Feature selection based on mutual information criteria of max-dependency, max-relevance and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell., 27: 1226-1238.

Robnik S.M. and I. Kononenko, 2003. Theoretical and empirical analysis of relief and relief. Mach. Learn., 53: 23-69.

Roweis, S.T. and L.K. Saul, 2000. Nonlinear dimensionality reduction by locally linear embedding. Science, 290: 2323-2326.

Sanchez, M.N., B.A. Alonso and S.M. Tombilla, 2007. Filter Methods for Feature Selection-A Comparative Study. In: Intelligent Data Engineering and Automated Learning. Hujun Y., P. Tino, E. Corchado, W. Byrne and X. Yao (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-540-77225-5, pp: 178-187.

Scholkopf, B., A. Smola and K.R. Muller, 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput., 10: 1299-1319.

Schreiber, S., J.M. Fellous, D. Whitmer, P. Tiesinga and T.J. Sejnowski, 2003. A new correlation-based measure of spike timing reliability. Neurocomputing, 52: 925-931.

Shaghaghi, M. and S.A. Vorobyov, 2015. Subspace leakage analysis and improved DOA estimation with small sample size. IEEE. Trans. Signal Process., 63: 3251-3265.

Song, L., A. Smola, A. Gretton, J. Bedo and K. Borgwardt, 2012. Feature selection via dependence maximization. J. Mach. Learn. Res., 13: 1393-1434.

Stoer, M. and F. Wagner, 1994. A Simple Min Cut Algorithm. In: Algorithms. Leeuwen, J.V. (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-540-58434-6, pp: 141-147.

Talavera, L., 2005. An Evaluation of Filter and Wrapper Methods for Feature Selection in Categorical Clustering. In: Intelligent Data Analysis. Famili, A.F., N.K. Joost, M.P. Jose, A. Siebes and A.D. Feelders (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-540-28795-7, pp: 440-451.

Watkins, A., J. Timmis and L. Boggess, 2004. Artificial immune recognition system (AIRS): An immune-inspired supervised learning algorithm. Genet. Program. Evol. Mach., 5: 291-317.

Yamada, M., W. Jitkrittum, L. Sigal, E.P. Xing and M. Sugiyama, 2014. High-dimensional feature selection by feature-wise kernelized lasso. Neural Comp., 26: 185-207.

Zeng, Z., H. Zhang, R. Zhang and Y. Zhang, 2014. A hybrid feature selection method based on rough conditional mutual information and naive Bayesian Classifier. ISRN. Appl. Math., 2014: 1-12.

Zhao, Z. and H. Liu, 2007. Spectral feature selection for supervised and unsupervised learning. Proceedings of the 24th International Conference on Machine Learning, June 20-24, 2007, Corvallis, Oregon, USA., ISBN: 978-1-59593-793-3, pp: 1151-1157.

Zhao, Z., L. Wang, H. Liu and J. Ye, 2011. On similarity preserving feature selection. IEEE Trans. Knowl. Data Eng., 25: 619-632.